A Robotic Approach to Understanding Robustness

Julien Hubert[1], Eiko Matsuda[2], Eric Silverman[3] and Takashi Ikegami[4]

[1,2,3,4]Department of General Systems Studies, The University of Tokyo, Tokyo, Japan
(Tel: +81-3-5454-4378; E-mail: {jhubert,eiko,erics,ikeg}@sacral.c.u-tokyo.ac.jp)

Abstract: Robustness is a property present in every living system which provides resilience against internal or external perturbations. Robustness is also highly desirable in engineered systems, as it makes them more resistant to unpredicted events. Despite its ubiquity, this concept is not yet understood and no existing framework provides a methodology to quantify it. Our work presents an approach to this problem through the use of robots acting as models for the study of robust organisms. Using robots, we look at how a change of robustness in sub-systems influences the robustness of the whole system. Our results show that using robotics offers an adequate level of complexity to study robustness while providing enough control to improve our understanding of this concept.

Keywords: Robustness, Hebbian Learning, Robot, Phototaxis, Phonotaxis

1. INTRODUCTION

Robustness can be defined as a property that allows a system to maintain its functions against internal and external perturbations[5, 7]. This property is present in biological systems and is highly desired in engineered systems. A simple example of how important robustness is for biological systems is mutational robustness[9]. In biological systems, natural evolution acts through mutations of genes. Some mutations can be beneficial but others are not. To protect living systems from the effect of detrimental mutations, robust systems have evolved so that mutations in their genotypes do not necessarily produce a change in phenotype.

In engineering, robustness is also an important factor when designing a complex system. If, for example, an airplane were to stop flying because of a variation in its environment such as a pressure change or a drop in temperature, the results would be quite problematic indeed.

Robustness is associated with two other concepts called structural stability and homeostasis[4]. Both of them are concerned with the maintenance of some states of the system and differ in the nature of the perturbations they target and the mechanisms they use to protect it. A system is considered stable if it can absorb small perturbations while retaining qualitatively similar dynamics. Homeostasis refers to mechanisms maintaining the states of a system. Robustness differs from stability and homeostasis, as the later maintains the states of the system while the former is concerned with maintaining its functions. Nevertheless, those notions can be combined to obtain robustness at a high level of the system.

For both biologists and engineers, understanding how a system acquires this property of robustness is an important task, but also a difficult one. Robustness is achieved through different mechanisms in biological systems which makes it difficult to isolate general principles that would allow engineers to develop highly reliable systems. Also, robustness is not a property that is given for free. An increase in robustness is generally associated with an increase in resources used by the system as well as an increase in its fragility, which raises the question of how it can be measured successfully.

Biologists have an important role to play in understanding robustness as it is present in all biological systems. Yet, a living system cannot be manipulated as easily as an engineered one which allows complete access to its inner workings. As such, any observation about the mechanisms of robustness might be influenced by hidden variables in the environment. A solution to this problem would be to study robustness by looking at engineered systems displaying this property; this would grant the researcher complete access to their inner workings and offer the possibility of manipulating these mechanisms. Unfortunately such engineered systems most frequently exist in a restricted or controlled environment and do not display the same kinds of behaviour as biological systems. Furthermore, engineered systems are developed with a specific function in mind while biological ones do not display such specificity.

The question we explore in this paper is how to compute the robustness of a system by observing the robustness of its sub-systems. As mentioned earlier, increasing robustness leads to increased fragility. This means that every robust system has its weaknesses. For systems acting in uncontrolled environments, testing for every perturbation that could present itself is an impossibility, and thus the possibility of a failure is always present. Our work takes the approach of looking at sub-systems that are less complex and easier to evaluate in order to get a better understanding of the robustness of the whole system. In this paper we present our initial study of a toy robotics experiment which we believe is a good testbed to study robustness. In section 2 we go deeper into the concept of robustness and we detail how robotics can help answering some of the questions surrounding it. Our experimental setup is explained in section 3 and our results presented in section 4. Finally, those are discussed in section 5.

2. UNDERSTANDING ROBUSTNESS THROUGH ROBOTICS

In recent years, robots have been used more frequently for the study of biological systems. Among many examples, they have been applied to the study of specific animals such as the cricket[8] or to understand the mechanics of more general concepts such as social behaviour[2]. Robots act as ideal models for living systems as they are situated at the boundary of biology and engineering. Being engineered systems, they provide complete access to their inner workings, but as they are embodied in the world, their behaviour cannot be predicted solely based on their inner workings. For those reasons, using robots to study robustness could benefit both biology and engineering.

Until now, robustness has not been the center topic of research in robotics. Many studies mention robustness, but only as a property assigned to the system they present rather than as the main object of study. In these studies, quantification of robustness has been achieved through a comparison of the performance of multiple controllers in a specific task. If a change in the controller increased the performance, this was considered an increase in robustness. Unfortunately, no work has given a complete account of the robustness of their controller outside the realm of their own experience. This is problematic as robustness is not a property that can be increased without consequences.

Trade-off is a concept often associated with robustness. In [1], the authors hypothesised that a variation of robustness in a system leads to additional changes that can be detrimental. This can take the form of a degradation of performance, an increase in resource demands or an increase in the fragility of the system[5]. In robotics experiments aiming at improving the robustness of a system, trade-offs are not considered and only the robustness against a specific perturbation is used for comparison. Based on the concept of trade-offs, however, the more robust system should have additional weaknesses that are not studied. As such, it is difficult to be definitive about which system is more robust outside of the limits of the experiment. Taking into account these trade-offs in the evaluation of robustness would require a system to be tested against all kinds of perturbations. This is not realistic as it is difficult to predict every situation in which a system could fail.

While it is difficult to compute the robustness of a system against all possible perturbations, focusing on sub-systems may make this task easier. Robustness is a property that is attached to the function of a system. Thus, the level of robustness is closely related to the performance of the system for that particular function. While a system might have many functions, sub-systems have a smaller set of functions due to their increased specificity. As such, those sub-systems would be easier to evaluate. The main problem with this approach is that it has not been shown how the robustness of a sub-system relates to the robustness of the whole.

With biological systems, experimenting with this approach is difficult as the concept of function does not transfer to them as well as for engineered systems designed with a specific goal in mind. Manipulating biological entities to single out sub-systems is also problematic, as they generally rely on the correct functioning of other sub-systems. Those limitations can be lifted by the use of robotic experiments to study robustness. Our approach is to create a system composed of multiple sub-systems for which the robustness can be independently evaluated. Using that knowledge, and by studying the robustness of the system as a whole, we hope to understand how the robustness of sub-systems influence the higher ones.

3. METHODOLOGY

This section explains the specifics of our experiments. Despite these studies taking place in both a real environment and a simulated one, the following explanations remain applicable for both situations. When differences exist, they are mentioned in the appropriate section.

3.1 The Task

Our experiments use a robot whose task is to move toward a target area in its environment. The robot moves in an open environment where one light source and one sound source are located at the same position. Facing those sources is a grid of 7x7 cells(see figure 1). Those are used as starting positions for the robot. The task of the robot is to navigate its environment using its sensors until it reaches a point at a maximum distance of 1 grid cell from the sources. The robot has 6 minutes to complete this task. If the robot reaches the goal within that time period, the trial is considered a success; otherwise, the trial is counted as a failure.

The light is generated by a white neon bulb emitting toward the grid. The emitted light cannot be perceived by the robot if it is located behind the light source. The sound is generated using white noise and can be heard from any direction in the environment. Figures 2 and 3 show respectively the light and the sound perceived by the robot at the different positions in the grid.

Our experiments were performed in three stages. In the first stage, the robot used only the light source to reach the goal(L). In the second stage, it used only the sound source(S). The last stage allows the robot to use both sources to locate the goal(LS). Each of these stages required specific implementations of the robot's controller that are detailed later in this section.

The performance of the robot is measured by its capacity to reach the goal in less than 6 minutes. For a single trial, the performance of a robot would be one if it reached the goal and zero otherwise. The duration of the trial is not considered in the performance measure. We relate this measure of performance to the robustness of the robot and will use both terms to describe its capacity to reach the goal.

For the S and LS conditions, different levels of noise are added to the sound sensor in order to to evaluate how the perturbation of one modality impacts the overall performance of the system. The noise is added by adding a value drawn from a uniform distribution to every measure reported by the sound sensor. The values for the noise presented in the results are always positive and represent the maximum MAX added. The range of those values is $[-MAX; MAX]$.

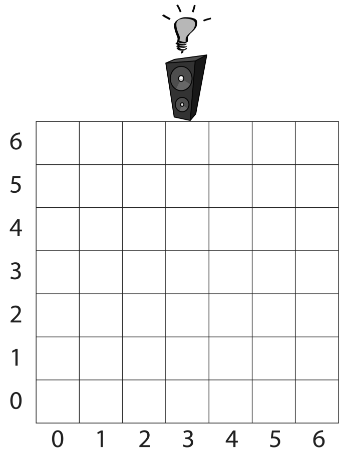

Fig. 1 Experimental arena. Each cell of the grid is a possible starting position for the robot.

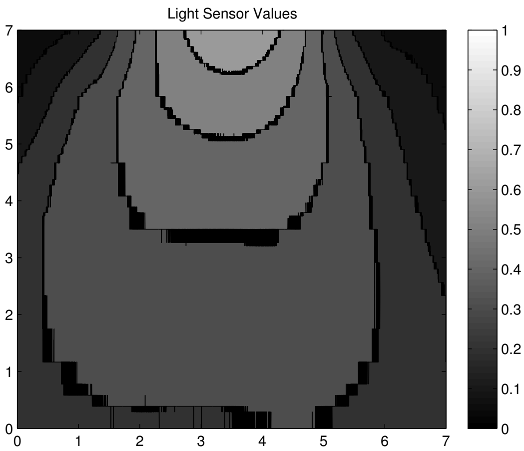

Fig. 2 Robot's perception of the light.

3.2 The Robotic Platform

Our robotic platform is the Lego Mindstorms NXT(see figure 4), which is a modular robot assembled from many elements such as motors, sensors and structural modules. The onboard processor is called the NXT and is a general purpose microprocessor whose function is to command the motors, retrieve information from the sensors and run custom software. The setup used in our experiments consists of two motors, one light sensor and one sound sensor. The two latter sensors measure the intensity of light and the volume of sound respectively.

Our experiments have been performed on the real platform but also in simulation. The latter has been designed using real sensor readings from the robot to reduce the discrepancies between the real environment and its simulated counterpart.

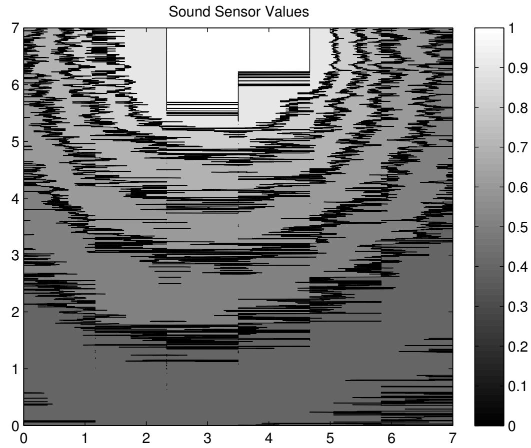

Fig. 3 Robot's perception of the sound.

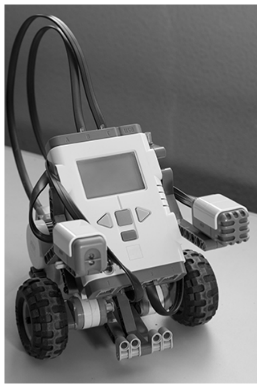

Fig. 4 The Lego Mindstorms platform

3.3 The Controller

The controller of the robot is a feedforward artificial neural network(NN) which possesses 4 or 7 inputs based on the number of sources it must track and 4 outputs. No hidden neurons are present. The input and output neurons are fully interconnected, and the weights are tuned using Hebbian learning[3].

The inputs of the NN are pre-processed in order to obtain binary inputs. The pre-processing is necessary for the Hebbian learning to be stable and is different for each type of source in the environment. The outputs are squashed to a range of $[0; 1]$ using the sigmoid function.

Before explaining the pre-processing, it should be noted that the robot is using two timescales to accomplish its task. The first one is the microprocessor timescale(MT) which corresponds to one step of the sensory-motor loop. One MT timestep involves one update of the sensors and of the motors. The second timescale is the neural timescale(NT) which corresponds to an update of the outputs of the NN.

The pre-processing applied on the inputs for the sound seeking task is as follows:

1. Input 0 is set to 1 if the current sound volume is higher than the volume measured 30 MT timesteps ago to which is added a small value of 0.03.
2. Input 1 is set to 1 if the current sound volume is lower than the volume measured 30 MT timesteps ago from

which is subtracted a small value of 0.03.
3. Input 2 is set to 1 if the current sound volume is greater than or equal to the volume measured 30 MT timesteps ago.
4. Input 3 is set to 1 if none of the other inputs is activated.
The controller goes through this list until one input is activated. The remaining ones are set to zero.

For the light seeking behaviour, the same system of rules is used but it is necessary to add an additional variable allowing the controller to distinguish between the ambient light in the room and the light coming from the goal. This is not necessary in the case of the sound as the room is quiet during the experiments. This memory, referred to as *imprint*, is updated every 120 MT timesteps and contains the intensity of the light at the time of its update. Every subsequent reading of the sensor is offset by this value. The following list details how the inputs are updated:
1. Input 2 is set to 1 if the intensity of the light is lower than a threshold set to 0.01
2. If the current intensity is lower than the intensity 10 MT timesteps ago minus a small value of 0.01, then there are two choices:
  (a) Input 1 is set to 1 if the robot goes backward.
  (b) Input 2 is set to 1 otherwise.
3. If the current intensity fits in none of the above, input 1 is set to 1 if the robot goes backward and input 0 is set to 1 otherwise.
The need to test for the direction of the robot arises from the unidirectionality of the light sensor which only picks up light when facing the source directly. In that sense, going backward is not necessary but can nevertheless happen in the early stages of the learning process. Because of that possibility, it is necessary to allow the robot to reverse its direction. The memory of 10 MT timesteps used for the light differs from the 30 MT timesteps of the sound. Both values have been determined experimentally in order to improve the performance in a real world environment. The sound being more noisy, a value of 30 MT timesteps is necessary to ensure a correct evaluation of the tendency of the robot to approach it. The light shows less noise and only requires 10 MT timesteps.

When the task combines light and sound, the robot uses a controller with 7 input neurons to allow it to combine both behaviours. In this condition the inputs are set according to the above algorithms. This means that at each NT timestep two inputs will be activated simultaneously: one for the sound and one for the light.

The NN has always 4 outputs regardless of the task. These represent the 4 possible behaviours that can be activated by the robot. To determine which behaviour is activated, a winner-takes-all strategy is used and the output with the highest activation wins. The four behaviours are:
1. Output 0 maintains the current behaviour.
2. Output 1 inverts the current behaviour. If the robot is going forward, it will go backward at the next MT timestep and vice-versa.
3. Output 2 modifies the current behaviour to create a left turn while maintaining the same direction.
4. Output 3 modifies the current behaviour to create a right turn while inverting the current direction.

The weights connecting the inputs to the outputs are tuned through Hebbian learning. Hebbian learning is an unsupervised learning algorithm which relies on correlations between inputs and outputs to decide if their connecting weights should be increased or not. There are many different implementations of Hebbian learning with different capabilities. The one we chose in our experiments is Oja's rule[6]. This rule implements the regular Hebbian learning while stabilizing the growth of the weights.

Despite the unsupervised nature of Hebbian learning, we cannot expect the NN to converge to the desired behaviour without guidance; Hebbian learning merely increases a weight when its input and output are simultaneously activated. In order to teach the network the task, we must manipulate the outputs before applying the learning algorithm in order to reflect what the correct behaviour should be based on which inputs are activated. When the experiments have either light or sound sources, the output representing the adequate behaviour receives an activation of 1 while the others 0. If both light and sound are used, two outputs can be activated. If both set of inputs point to the same output, it receives an activation of 1. If two different outputs are selected, they both receive 0.5. This scheme has been implemented to promote the cooperation of both sub-systems.

4. RESULTS

Our results are based on three sets of experiments done in a simulated environment: a light only condition(L), a sound only condition(S) and a light and sound condition (LS). Every starting position of the robot in the arena was tested 1000 times and the success rate was averaged to obtain the results presented in this section.

The performance for each condition is shown in figure 5. In this figure L remains constant but the other conditions vary following the amount of uniform noise added on the sound. We can observe that the performance of S and LS increase to a maximum located around 0.05 of noise then decreases progressively with the increase of the noise. Initially, the performance of S is higher than LS but this changes when the noise rises above 0.2. At that point, the performance of LS is higher than S. At any moment, the performance of S and LS are both inferior to L.

From the performance of the robot in the three conditions, it is clear that the merging of L and S is not producing a performance linearly related to their individual efficiency. We also see that the behaviours are not naturally mixed to compensate for each other's limitations. Instead, we see that the performance of LS is much lower than L. To understand why this is so, comparing the performance of S, L and LS at each starting point on the grid can be helpful. Figure 6 presents this information for L.

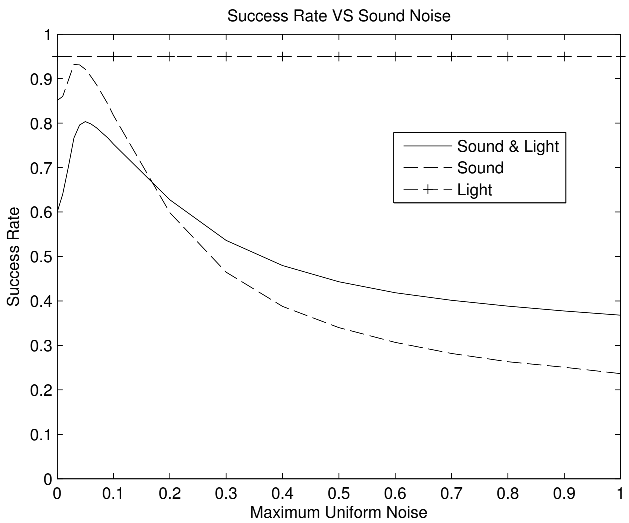

Fig. 5 Performance in each condition with varying uniform noise added to the sound.

We see that the performance is at its maximum on the left side of the arena and degrades toward the right. In figure 7, a comparison of S and LS is shown for noise levels of 0, 0.05, 0.2, 0.5 and 1.0. With low levels of noise, the performance of S close to the source is very high. When there is no noise, the performance is above 80% over almost the entire arena. For LS, however, the picture is different: we still see high performance but only until the middle of the arena, and the robot does not manage to maintain it over the whole width. This explains why the performance of LS is lower than S at low noise levels as the area of high performance for S is much bigger than the one for LS. With the noise increasing, the performance of S decreases and the zone of high performance reduces progressively around the sound source. This decrease also appears in the case of LS but the speed of this deterioration is lower than for S. At equal noise levels, the area of high performance for LS becomes higher than the one for S. This explains why, at a level of noise above 0.2, LS is performing better than S as the noise has a bigger impact on S than on LS.

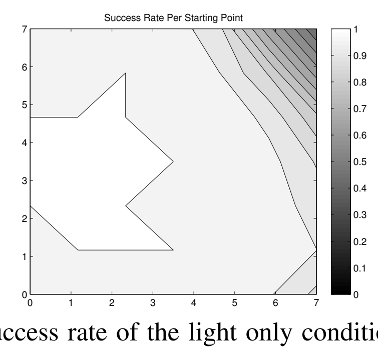

Fig. 6 Success rate of the light only condition over the arena.

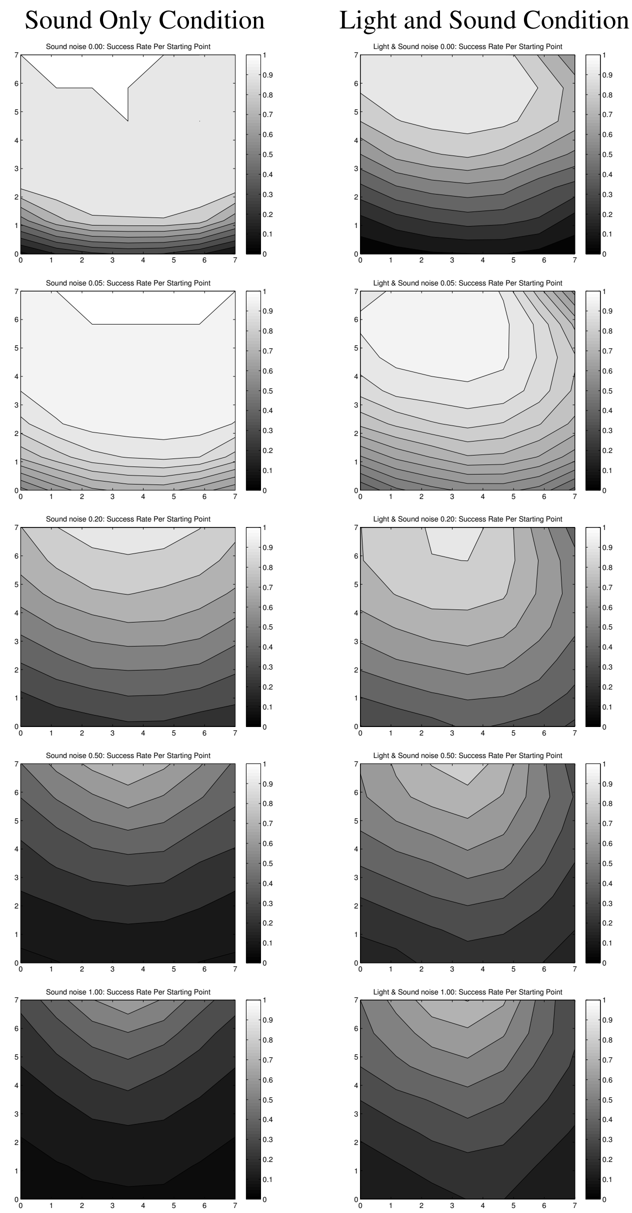

Fig. 7 Comparison between the sound only and the light and sound conditions. From top to bottom, the figure shows the success rate for each starting position with a uniform noise level respectively equal to 0, 0.05, 0.2, 0.5 and 1.0.

5. CONCLUSION

The aim of this work was to study how the robustness of a system can be explained by studying the robustness of its sub-systems. With that purpose in mind, we created an experiment where a robot has to reach a goal indicated by a light and a sound source. The behaviour of the robot was implemented by a mixture of pre-processing on the sensor values with a neural network tuned through Hebbian learning. We tested this robot in 3 environmental conditions: light only, sound only and both sources present.

Despite our results being preliminary, they clearly show that the interaction of the two sub-systems does not present any simple relation to the robustness of the complete system. Despite the light sub-system being highly efficient in reaching the goal, its contribution is not visible in the system's behaviour. The sound sub-system appears to be driving the performance of the system. Nevertheless, given that the performance of the latter sub-system decreases faster with uniform noise than the complete system, it is clear that the light sub-system offers some additional robustness.

Our future work should focus on completely understanding the relationship between the sub-systems and the full system. This knowledge would allow us to uncover the parameters driving this relationship and to draw a first hypothesis on how to compute the robustness of a system from measurements done on its sub-systems. This could then be tested by adding noise on the light sensor in our simulation and predicting the robustness of the system. The final step would be to test our hypothesis on the real robot.

The important message of this work is that, even with a simple setup and a controlled environment, understanding how robustness exists at different levels of a system is not a straightforward task. Carrying out this same experiment using a biological system would be impossible. For that reason, robotics can be a useful method for the study of robustness.

ACKNOWLEDGMENTS

This work has been partially supported by a Grant-in-Aid for Scientific Research on Priority Areas "Emergence of Adaptive Motor Function through Interaction between Body, Brain and Environment" from the Japanese Ministry of Education, Culture, Sports, Science and Technology. J.H. thanks the Monbukagakusho Scholarship from the Ministry of Education, Culture, Sports, Science and Technology.

REFERENCES

[1] J. M. Carlson and J. Doyle. Highly optimized tolerance: A mechanism for power laws in designed systems. *Phys. Rev. E*, 60(2):1412 – 1427, 1999.

[2] D. Floreano, S. Mitri, S. Magnenat, and L. Keller. Evolutionary conditions for the emergence of communication in robots. *Current Biology*, 17(6):514–519, March 2007.

[3] D. O. Hebb. *The Organization of Behavior*. Wiley, New York, 1949.

[4] E. Jen. Stable or robust? What's the difference? In Erica Jen, editor, *Robust Design: A Repertoire of Biological, Ecological, and Engineering Case Studies*. Oxford University Press, 2005.

[5] H. Kitano. Towards a theory of biological robustness. *Molecular Systems Biology*, 3(137), 2007.

[6] E. Oja. Simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15(3):267–273, November 1982.

[7] E. Silverman and T. Ikegami. Robustness in artificial life. In *Proceedings of Complex '09 Conference*, Tokyo, Japan, 2009.

[8] B. Webb. Can robots make good models of biological behaviour ? *Behavioral and Brain Sciences*, 24(6):1003–1050, 2001.

[9] C. O. Wilke and C. Adami. Evolution of mutational robustness. *Mutation Research*, 522(1-2):3–11, January 2003.